EAC Toolkit - Instructor Module for Theory Building Activities: Mountain Terrorist Exercise

REFERENCE OR LINK TO STUDENT MODULE

Introduction
The Mountain Terrorist scenario that constitutes the core of the corresponding student module comes from the philosopher, Bernard Williams. It is common in introductory ethics textbooks (such as Geoffrey Thomas' An Introduction to Ethics). Williams' own account can be found in several anthologies including **Ethical Theory: Classics and Contemporary Readings, 5th edition** (2007) edited by Louis Pojman. (See note below,) The corresponding student module uses the core scenario to introduce students to ethical argument, to get them to recognize that they are already employing ethical arguments, and to get them to practice the virtue of reasonableness.
Core Instructor Module Links include...

- The Instructor Module Template which provides the general framework for instructor modules in the EAC Toolkit.
- The student module, "Theory Building Activities: Mountain Terrorist Exercise," which is published in the Connexions® Content Commons.

    - The student module is also accessible in the course, **Corporate Governance** (col10396). A link included in this module provides access to this course.
    - The student module can be accessed through the course, **Professional Ethics in Engineering** (col10399) which is published in the published in Connexions® Content Commons and linked to in this instructor module.
    - See notes below for textbooks that present the core dilemma scenario.

INSTRUCTOR RESOURCES(Sharing Best Practices in EAC!)

This section contains information related to the above referenced Student Module. The intent and expectation is that the information

contained in this section will evolve over time based on the experiences and collaborations of the authors and users of the Student Module and this Instructor Module. For example, the authors, collaborators or users can provide the following kind of information (mainly directed at or intended for instructors).

Module-Background Information

Where did this module come from? (e.g. A workshop, news story, based on a movie, etc.) What condition is it in? (e.g. first draft, needs editing, publishable, etc.) How has it been used in the past? (e.g. in classroom, workshop activity, ethics debate, etc.) Other relevant or interesting details

The first time this module's author became aware of its use in the classroom was in a workshop on Agriculture Ethics led by Paul Thompson, then of Texas A and M University, in 1992. Thompson's particular instantiation of this exercise was broadcast over the AG-SAT network in the spring of 1992 during a course on Agricultural Ethics. The module is based on a scenario, "Jim and the Jungle," first put forth by Bernard Williams (see note below) in a work devoted to the criticism of utilitarianism. While the scenario does present challenges to utilitarianism (and deontology), it is used in this context to help students see how ethical theories are encapsulated in moral reasoning and moral arguments.

Learning Objectives

- **Ethical Reasoning**: Practicing and improving ethical reasoning.
- **Ethical Evaluation**: Evaluating decision alternatives in terms of their ethics.
- **Ethical Awareness**: Becoming aware of how ethical theory and ethical issues are embedded in everyday discourse.
- **Reasonableness**: Practicing the virtue of reasonableness in the context of reasonable disagreement

- Learning **how to recognize** the ethical theory embedded in everyday reasoning.

The following table documents the objectives for the student module.
EAC Matrix
https://cnx.org/content/m14351/

> This table documents an
> EAC integration activity
> in terms of moral
> objectives, accredition
> criterion, and curriculum
> location.

Instructional / Pedagogical Strategies

Which pedagogical or instructional strategies are used or suggested for this module. (For example: Discussion/Debate, Decision-Making Exercise, Presentation, Dramatization or Role Playing, Group Task, Formal or Informal Writing, Readings, among others)
This module employs the following pedagogical strategies:

- **Informal Writing**: Students prepare for the activity by reading the scenario and writing out their reponse.
- **General Class Discussion**: Students discuss the scenario as a class. No attempt is made to reach agreement or closure.
- **Cooperative Learning**: Students are divided into groups of three to five. Each student reads his or her written response to the other group members. Then the group is charged with reaching an agreement within a time frame or outlining the terms of their disagreement
- **Eliciting Knowledge**: The instructor provides a formal debriefing outlining the ways in which students have used ethical reasoning, the structures in terms of which they raised and resolved disagreements, and how they made use of different ethical theories in formulating

their justifications and arguments. If time permits, the instructor can add a more formal introduction to different kinds of ethical theory that draws the theory from the arguments the students have made during their discussions.

- This module is used to introduce two new modules, the Ethics of Team Work and Moral Exemplars. (The latter explores issues in ethical leadership.)

For those new to teaching cases and teaching by discussion, the Computing Cases website has information and links that will be of great help. Address: http://computingcases.org/general_tools/teaching_with_cases/teaching_w_cases_intro.html

Assessment / Assurance of Learning

What assessment or assurance of learning methods are used or suggested for this module? (For example: 1-minute paper, Muddiest Point, Quiz/Test Items, Oral Presentation, Student Feed-back, among others). What did or didn't work?

Modes of Assessment

1. **Informal Writing**: A baseline for assessment can be established by examining the students' initial written responses to the scenario. For example, student responses can be assessed in terms of where the responses provided by the students fit on Kohlberg's scale of moral development. In this particular version, students are assessed in terms of the moral schemas that are triggered by the dilemma situation. (See Rest et al below.)
2. **Muddiest Point Exercise**: The student module can also be assessed by using a simple Muddiest Point exercise that asks the students to indicate the strongest and weakest (=muddiest) parts. (See figure just below for handout.)
3. **EAC Module Assessment Form**: A form modified from one developed and used by Michael Davis of IIT helps provide a more detailed assessment of this and other modules. See figure below.

Muddiest Point Handout

This file provides a handout in Word form for carrying out a Muddiest Point assessment activity. This module's author learned about this activity from an assessment webpage at Southern Illinois University at Edwardsville. For more information consult the Muddiest Point link in this module.

Module Assessment Form

This file contains an assessment form developed by Michael Davis of the Illinois Institute of Technology to assess EAC integration projects. It has been slightly modified by the authors.

Preliminary Assessment Results from Muddy Point Exercise

1. Some students felt constrained by the dilemma framing of the scenario. They didn't like being forced to choose between shooting a villager or walking away. They wanted more freedom to explore other options.

2. Other students wanted the scenario to provide more details to aid them in making their decision. For example, did the villagers collaborate with the enemy, which ones collaborated, what was the cause of the terrorists, etc. They felt this would make it easier to make an defend an ethical choice.
3. Some students (not all) had trouble seeing how their modes of reasoning made use of established ethical modes of argument.
4. Finally, many wanted to see more closure in the activity. For example, what did the instructor think, what was the correct answer to the dilemma, how did this relate to their project study in business ethics.
5. As a result of this assessment exercise, a new conclusion was added to the student module. It emphasized how moral theory was embedded in the students' comments and how the students practiced the virtue of reasonableness in listening to different positions and searching for areas of agreement.

Pedagogical Commentary

Any comments or questions regarding this module? (For example: suggestions to authors, suggestions to instructors (how-to), queries or comments directed o EAC community, pitfalls or frustrations, novel ideas/approaches/uses, etc.)

This exercise always evokes a strong response from students. In final course evaluations, students often refer to this exercise as the most memorable experience in the course. But many are frustrated by the lack of closure and are uncomfortable with the lack of closure. The following list provides a partial set of guidelines to keep in mind when teaching this module:

- The most important thing an instructor can do in this module is listen. Students often make use of moral arguments and ethical theory. Listening carefully to their arguments and highlighting how they use argument and theory provides a means of introducing ethical theory without falling prey to the theory-practice gap.

- Closure can be reached by having students reflect on how they dealt with disagreements with their peers. In small groups, for example, students who have trouble agreeing can be asked to reflect on this experience. They can be encouraged by showing them how their discussion, while not issuing in agreement, often sharpened and clarified the nature and terms of disagreement.
- Students often come into an ethics class with the idea that all ethical problems are dilemmas, that is, forced choices between two, equally bad alternatives. The frustration they experience in resolving the Mountain Terrorist dilemma can be used to motivate them to reframe problems that initially take the dilemma form. In other words, the exercise can be used as an occasion to introduce and practice moral imagination.

Appendix (Annotated)

Bibliography

1. Bernard Williams, "Against Utilitarianism," in **Ethical Theory: Classics and Contemporary Readings, 5th edition** (2007) edited by Louis Pojman, Belmont: Wadsworth: 219-228.
2. Geoffrey Thomas (1994) **An Introduction to Ethics**, U.K.: Oxford.
3. James R. Rest, D. Narvaez, M.J. Bebeau, and S.J. Thoma. (1999) **Postconventional Moral Thinking: A Neo-Kohlbergian Approach**, Lawrence Erlbaum Press, Hillside, N.J.
4. Mark Johnson (1993) **Moral Imagination: Implications of Cognitive Science for Ethics**, University of Chicago Press, Chicago.
5. Anthony Weston (2006) **A Practical Companion to Ethics**, U.K.: Oxford University Press.

Additional information or annotations for instructors regarding the Student Module Appendix

Previewing a Module
This module explains how to preview a module before publishing.

After editing the module, you should preview the module to verify that it appears as you want it to appear. You can view modules from the "Preview" tab in the following ways:

- Preview the module on-line
- Preview the module in a print version (PDF file)
- View the change summary

Note that these three options are also available in the Module Status sidebar. The sidebar also has a few extra options that are not available on the "Preview" tab (`Source` and `Discard`) which we discuss below. [missing_resource: /content/m19610/latest/module-status.png]

The Module Status portlet.

Previewing Modules (Online)

To view a module online, use the following steps:

1. Click on the "Preview" tab. The "Preview" screen displays.
2. Click the `Online Preview` link. This displays an on-line version of the module that is similar to the on-line display of the published module. The difference between the preview and the published versions is that the preview cannot display certain porlets (like the Recently Viewed portlet). Review the module content and verify that it appears as you intended it to display.
3. Use your browser's `Back` function to return to the "Preview" screen.

Previewing Modules (PDF)

The print version of a module is formatted differently than the online version. This difference is necessary because of the differences in the online

and printed media. To view a print version of the module, use the following steps:

1. Click on the "Preview" tab. The "Preview" screen displays.
2. Click the `Print Preview` link. An "Opening index.pdf" dialog box appears.
3. Verify that the option to open the file with a PDF viewing application is selected and click `OK`. The print version of the module displays.
4. Review the module display and verify that text and figures display as you intended them to display.
5. You can print this PDF file using your PDF viewer's file/print option.
6. Close the PDF viewing application.

Viewing Change Summary

When editing an existing module, you can view a summary of the differences between the last published version of the module and the version you are currently editing. Click the `Changes` link to view the change summary. This option is useful when you checked out a module some time ago and do not remember what changes you made to it.

[missing_resource: content/m19610/latest/module-view-changes.png]

The "Changes to module" summary.

The "Changes to module" summary lists changes to the module, including role assignments and the names of any files that were changed, files that were added, and files that were removed. Click on a file name to display a brief description of the changes.

Clicking on the `index.cnxml` link will display the CNXML markup of the module. Lines of markup which have been removed or the "before" version of altered lines are shown with a minus sign in the left margin.

Lines which have been added or the "after" version of altered lines are shown with a plus sign in the left margin.

[missing_resource: /content/m19610/latest/module-view-diff.png]

An example of the "diff" log, showing the changes in index.cnxml.

Previewing the code source

Occasionally it may be convenient to view the markup source for a module. To view the source of the module, click the **Source** link under the "View" heading in the "Module Status" sidebar. The most recent markup source that has been saved displays with line numbers.

Note: This feature is useful when you are trying to find a markup error.

Discarding Your Changes

The situation may arise in which you wish to return a module that you are editing back to its original state: the state in which it was before you began editing it. To do this, click the **Discard** link under the "Actions" heading in the "Module Status" sidebar.

[missing_resource: content/m19610/latest/discard-module.png]

- **Discard**The module will revert back to the state it was in when you checked it out. If the module is new and has never been published, you will lose all the content you entered and the module will be blank.
- **Delete**This button removes the module from your current work area but **does not** remove it from the repository. Once published, a module remains in the Connexions repository.
- **Cancel**This button will return you to the screen you just left.

Editing Module Files
This module explains how to upload associated files to a module.

It is possible to create a module with just textual information. In this case, the content is all included in the CNXML file of the module. But in some cases you may want to include associated files with your module content, like an image, video, PDF, PPT, or other file.

You can upload associated files on the "Files" tab at the top of the module edit page. In this case, the module is really more like a "box", which can contain any number of files associated with the module in addition to the CNXML file itself. When you upload files, you create an instance of the file in Connexions that you can then link to from your module for readers to download, or even embed directly into your module.

This module will explain how to:

- Upload files to your module
- Link to files in your module
- Manage files on the "Files" tab

Uploading Files

File Guidelines

Before you add an associated file to a module, please note the following guidelines for files and filenames:

- The size of a file must be smaller than 50 MB.
- The first character in a filename cannot be a digit. For example, a file named "1965mustangphoto.gif" would result in an error. Renaming the file to "photo1965mustang.gif" prevents the error.
- A filename cannot contain an embedded blank space. For example, the file named "eiffel tower.png" would cause an error. Renaming the file to "eiffel_tower.png" or "eiffeltower.png" prevents the error.
- Use only alphanumeric, "." (dot), "-" (hyphen), and "_" (underscore) characters in a filename.

- [Multimedia files](#) must have a recognized MIME (Multipurpose Internet Mail Extension) type.

Uploading a file to a module

To add a file to a module, use the following steps:

1. Click on the ["Files" tab](#). This page shows a list of all included files in the module. The `index.cnxml` file exists in every module -- this is the CNXML file that creates the module contents which you edit on the `Edit` tab.
2. Click the `Add new image/included file` button to add a file.

[missing_resource: ../content/m19610/latest/module-files.png]

The "Files" tab with the "Add new image/included file" button.

3. Click `Browse` to navigate to the file in your system that you want to upload.
4. Click the blue `Upload` button to upload the file into Connexions. A confirmation message will display, along with a preview if the file is an image.

[missing_resource: ../content/m19610/latest/uploadNewFile.png]

The file upload page.

5. Click the `<<Return to file listing for module` link at the top of the page to return to a list of all files uploaded to this module. Here you can click the `Add new image/included file` button and start the process over to upload another file.

Uploading multiple files at once

If you have a lot of files that you want to include in a module, uploading them one by one can take a while. You can use the [ZIP importer](#) to import multiple files at once.

[missing_resource: ../content/m19610/latest/module-files-zipimport.png]

The zip importer imports multiple files at once.

Linking to files

Readers can download a file that you added in two ways.

1. By adding the file to the module's [Featured Links](#). This is probably the easiest way, but although the links appear online, they will not be included in the PDFs that are automatically generated.
2. You can add a link to the file within the module text. If the file you added was called "myfile.doc", you would add the following snippet to your module. "Download this `<link resource="myfile.doc">file.</link>`" More instructions on adding links in modules can be found on the [<link> eip-help page](#).

Note: You can also [embed](#) the file into the CNXML document using the CNXML`<media>` tags.

Managing Files

> **Note:Do not** remove or rename the
>
> `index.cnxml`
>
> file. This file holds the text and CNXML-markup of your module.

Removing Files

To remove a file from a module, select the checkbox next to the title of the file you want to remove and click `Remove`. You can select multiple files in the module and remove them in one operation.

Renaming Files

To rename a file in a module, use the following steps:

1. Select the checkbox next to the title of the file you want to rename and click `Rename`. The "Rename items" screen displays.
2. Enter the new name for the file in the "New File Name" field.
3. Click `Rename All`. You can rename multiple files at one time by selecting more than one item by clicking on the work area contents screen.
4. Update any existing links in your module to point to the new file name.

The files displayed on the "Files" tab are updated to reflect the changes.

Cutting or Copying Files

To copy and paste files in one module to a different module, use the following steps:

1. Select the checkbox next to the title of the files you want to cut or copy and click `Cut` or `Copy`. `Cut` removes the original item at the end of this process. `Copy` leaves the original item in its present location.
2. Display the location where you wish to paste the file, such as another module or work area, and click `Paste`.

EAC Toolkit - Instructor Module for UPRM Ethics Bowl Activity

REFERENCE OR LINK TO STUDENT MODULE

This Ethics Bowl Instructor Module corresponds to the student module, **EAC Toolkit - UPRM Ethics Bowl - IIT Summer Institute Follow-up** (see pre-requisite link on the right). The student module is part of the Corporate Governance course published in Connexions (col10396). First implemented as a capstone activity for engineering ethics classes (at the suggestion of Robert Ladenson of IIT who originated the Intercollegiate Ethics Bowl held at the annual meetings of the Association for Practical and Professional Ethics), this activity was reported on in its initial stages by Dr. Jose Cruz during an NSF-funded workshop on Ethics Across the Curriculum led by Michael Davis and carried out at the Illinois Institute of Technology in 2003. Since then, the activity has undergone several revisions. This module and the student module link to Dr. Cruz's report. But they also include material added and revised since this report. By collecting this material in the student and instructor modules, readers can see how the competition has evolved as well as learn how it can be adapted to different learning situations.

INSTRUCTOR RESOURCES (Sharing Best Practices in EAC!)

This section contains information related to the above referenced Student Module. The intent and expectation is that the information contained in this section will evolve over time based on the experiences and collaborations of the authors and users of the Student Module and this Instructor Module. For example, the authors, collaborators or users can provide the following kind of information (mainly directed at or intended for instructors).

Module-Background Information

Where did this module come from? (e.g. A workshop, news story, based on a movie, etc.) What condition is it in? (e.g. first draft, needs editing, publishable, etc.) How has it been used in the past? (e.g. in classroom, workshop activity, ethics debate, etc.) Other relevant or interesting details

Robert Ladenson describes the growth of the Ethics Bowl concept in his paper, "The Educational Significance of the Ethics Bowl. Currently, he directs an Intercollegiate Ethics Bowl consisting of regional competitions and a national competition held annually at the meetings of the Association for Practical and Professional Ethics. The ICEB has over the years developed prestige and stature including winning the American Philosophical Association prize for Excellence and Innovation in Philosophy Programs.

The Puerto Rican instantiation of the competition in Engineering and Corporate Governance classes represents something of a de-evolution of the concept. Ladenson began the competition within his school, the Illinois Institute of Technology; then it grew into its present form. At UPRM, we have brought the competition back into the classroom where it serves as the capstone activity for classes in Practical and Professional Ethics. With the minimal modifications we have made, it has turned into a very powerful classroom tool for teaching different aspects of Practical and Professional Ethics.

This particular version of the Ethics Bowl has gone through four stages.

- First, judges from Humanities and Engineering were invited to the class, and, ona Monday-Wednesday-Friday schedule within the confines of a 50 minute class, the entire competition took place and scores were calculated and announced. Each student team debated twice. But assessment results showed that students wanted more time to carry out each stage of the competition and they wanted more feedback from the judges.
- For this reason, the second phase of the competition was carried out during the longer class sessions of the Tuesday-Thursday schedule. While students had more time to formulate their arguments and

responses, they still asked for a more relaxed schedule that included more feedback from the judges.

- In the third phase, the debates were held outside the regular class schedule as determined by the students, usually on Saturdays and holidays. While this generally worked well for the students, it became difficult to find engineering and humanities faculty members willing to give up 6 to 8 hours of their weekend.
- In the fourth phase, two student debating teams compete during the regular Monday-Wednesday-Friday schedule. The first team defends its case in the first class period. The second receives and discusses its case in the following class period. Along with the two debating teams, two peer review teams serve as judges asking questions during the questioning period and scoring at the end of each class period. Finally, a third class period is given over to the peer review teams announcing and explaining their scoring. The advantage of this version of the competition is it solves both the time and feedback concerns that persisted through the prior instantiations of the debate.

The authors of this module have discussed issues concerning the integration of the Ethics Bowl into the classroom in a paper entitled, "The Ethics Bowl in Engineering Ethics at the University of Puerto Rico - Mayaguez. (Teaching Ethics, 4(2), Spring 2004: 15-32.) This paper discusses the assessment methodology used and summaries of the assessments of the first two years of the competition. After itemizing what the authors beieve are the considerable accomplishments of the classroom activity, it goes on to mention several ethics bowl challenges. Ethics bowl assessment has continued after the publication of this article. Two particular challenges have emerged: clarifying as much as possible the judging criteria and providing the debating teams as much constructive feedback as possible. This instructor module and the corresponding studentmodule describe ethics bowl innovations that attempt to respond to these assessment issues.

An article by Michael Davis, "Five Kinds of Ethics Across the Curriculum: An Introduction to Four Experiments with One Kind", discusses this classroom use of the Ethics Bowl as an instance of "professional ethics across the curriculum." In a footnote worth quoting, Davis distinguishes the Engineering Ethics Bowl held at UPRM from the Intercollegiate Ethics

Bowl that has come to form a central part of the yearly APPE meetings: "This description of the ethics bowl differs from Robert F. Ladenson, "The Educational Significance of the Ethics Bowl," Teaching Ethics 1(1) March 2001: 63-78, in at least three ways. First, it describes the process of transplanting the ethics bowl to a more or less non-English speaking environment. Second, it it desribes an effort to use the ethics bowl for professional ethics across the engineering curriculum (rather than, as Ladenson presents it, use it to do social issues across the curriculum). And third, it it describes the process of making the ethics bowl fit the time-constraints of an ordinary (engineering) classroom."

We add three further distinctions to Davis'.

- First, we have sought to use the ethics bowl as a way to generate feedback for students on their skills in ethical decision-making. Three classes are devoted to each competition. The third class provides an effective debriefing on the competition. It is not always easy for students to receive such feedback, but debriefing activities help them to interpret feedback and put it to good use.
- The ethics bowl provides an excellent opportunity for students to refine their understanding of what Rest terms "intermediate moral concepts." Examples of these concepts include "paternalism", "conflict of interest", "faithful agency", "public wellbeing", and "collegiality". By choosing cases that explore the boundaries of these concepts, the ethics bowl can be used as a wayy of proceeding from clear instances of these concepts to more problematic instances. This activity of prototyping forms an essential part of our coming to understand the thick, complicated moral concepts so essential to everyday moral reasoning.
- Studies like the Hitachi Report demonstrate that much of the moral decision-making that our students will be exercising will be shaped and constrained by the organizational environments in which they work. Companies built around financial objectives elicit one kind of moral advocacy while those built around customer- or quality-oriented standards require quite different strategies. With carefully chosen cases, the ethics bowl can recreate these environments to allow students to practice decision-making under real world constraints. The classroom becomes an "ethics laboratory".

Learning Objectives

What are the intended learning objectives or goals for this module? What other goals or learning objectives are possible?

Below are different lists of content and skill objectives of the ethics bowl. Not all of themapply at once. But they can be bundled together to fit different forms or instantiations. For example, a Corporate Governance ethics bowl would differ from an Engineering Ethics Bowl in terms of content objectives. This difference could be reflected in case selection, especially through the different basic and intermediate moral concepts covered by a case. The same would apply to a list of skill objectives; not all the UPRM skills could be covered in a given case or even a given competition. But a wide range of cases selected for student preparation could at least touch upon these skills.

Content Objectives come from the AACSB Ethics Education Task Force Report. In the Corporate Governance class (Connexions course, col10396), a special effort has been made to make the ethics bowl responsive to these content requirements.

Content Objectives

- **Ethical Leadership (EL)**: (a) "Expanding…awareness to include multiple stakeholder interests and…developing and applying…ethical decision-making skills to organizational decisions in ways that are transparent to…followers." (b) "Executives become moral managers by recognizing and accepting their responsibility for acting as ethical role models."
- **Decision-Making (DM)**: "Business schools typically teach multiple frameworks for improving students' ethical decision-making skills. Students are encouraged to consider multiple stakeholders and to assess and evaluate using different lenses and enlarged perspectives."
- **Social Responsibility (SR)**: "Businesses cannot thrive in environments where societal elements such as education, public health, peace and personal security, fidelity to the rule of law, enforcement of contracts, and physical infrastructures are deficient."
- **Corporate Governance (CG)**: (a) "Knowing the principles and practices of sound, responsible corporate governance can also be an

important deterrent to unethical behavior." (b) "Understanding the complex interdependencies between corporate governance and other institutions, such as stock exchanges and regulatory bodies, can be an important factor in managing risk and reputation."

UPRM Ethical Empowerment Skills List

- UPRM Objectives have been taken from SEE, 546-547:
- **Ethical Awareness:** "the ability to perceive ethical issues embedded in complex, concrete situations. It requires the exercise of moral imagination which is developed through discussing cases that arise in the real world and in literature."
- **Ethical Evaluation**: " the ability to assess a product or process in terms of different ethical approaches such as utilitarianism, rights theory, deontology, and virtue ethics." This skill can also be demonstrated by ranking solution alternatives using ethics tests which partially encapsulate ethical theory such as reversibility, harm, and publicity.
- **Ethical Integration**: "the ability to integrate—not just apply—ethical considerations into an activity (such as a decision, product or process) so that ethics plays an essential, constitutive role in the final results."
- **Ethical Prevention**: the ability to (a) uncover potential ethical and social problems latent in a socio-technical system and (b) develop effective counter-measures to prevent these latent problems from materializing or to minimize their harmful or negative impact. Ethical is an adjective that modified "prevention"; hence ethical prevention does not mean the prevention of the ethical.
- **Value Realization**: "the ability to recognize and exploit opportunities for using skills and talents to promote community welfare, enhance safety and health, improve the quality of the environment, and (in general) enhance wellbeing.

Hastings Center Goals

- Stimulate the moral imagination of students
- Help students recognize moral issues
- Help students analyze key moral concepts and principles
- Elicit from students a sense of responsibility

- Help students to accept the likelihood of ambiguity and disagreement on moral matters, while at the same time attempting to strive for clarity and agreement insofaras it is reasonably attainable
- (from Pritchard, Reasonable Children, 15)

Goals for ethical education in science and engineering derived from psychological literature (Huff and Frey)

- Mastering a knowledge of basic facts and understanding and applying basic and intermediate ethical concepts.
- Practicing moral imagination (taking the perspective of the other, generating non-obvious solutions to moral problems under situational constraints, and setting up multiple framings of a situation)
- Learning moral sensitivity
- Encouraging adoption of professional standards into the professional self-concept
- Building ethical community

The figure below provides an EAC Matrix used at the University of Puerto Rico at Mayaguez in the College of Business Administration. It also separates the objectives mentioned above into primary and secondary areas of focus. Finally, it imports information as to whether the actual outcomes meet the objectives.
Ethics Bowl Student Module Matrix
https://cnx.org/content/m14387/

This Matrix identifies the learning objectives of the corresponding student module by cross referencing the moral development objectives, accreditation criteria and the curricular "space" the the module fills.

Instructional / Pedagogical Strategies

Which pedagogical or instructional strategies are used or suggested for this module. (For example: Discussion/Debate, Decision-Making Exercise, Presentation, Dramatization or Role Playing, Group Task, Formal or Informal Writing, Readings, among others)
This module employs the following pedagogical strategies:

- **Informal Writing**: Students prepare their cases by writing short summaries.
- **Formal Writing**: After ethics bowl competition, students in teams prepare a formal, in-depth case analysis of the case they debated during the competition.
- **Cooperative Learning**: Students are divided into teams to prepare for debate, carry out debate, peer review as judges other debates, and prepare an in-depth follow-up analysis. The also prepare preliminary and final self-evaluations to assess the effectiveness of their work together as teams.
- **Pre-Debate Skills**: Theethics bowl requires considerable preparation. Students need practice with ethical and practical frameworks as well as work on researching cases and working with the basic and intermediate moral concepts posed in the cases. Students also need an orientation to the competition that includes the rules, time line, and debating and presenting strategies. Finally, it is important to explain carefully to students the ethics bowl scoring criteria.

Assessment / Assurance of Learning

What assessment or assurance of learning methods are used or suggested for this module? (For example: 1-minute paper, Muddiest Point, Quiz/Test Items, Oral Presentation, Student Feed-back, among others). What did or didn't work?

The figures below provide handouts for assessing this module. The Ethics Bowl scoring sheets contained in the Student Module also provide excellent means for assessing this activity.

Muddiest Point Assessment Form

https://cnx.org/content/m14387/

The attached word document provides a handout to assess this module in terms of its weakest and strongest points.

Module Assessment Form

https://cnx.org/content/m14387/

This figure contains an assessment handout, a modification of a form developed by Michael Davis for IIT EAC workshops.

Pedagogical Commentary

Any comments or questions regarding this module? (For example: suggestions to authors, suggestions to instructors (how-to), queries or comments directed o EAC community, pitfalls or frustrations, novel ideas/approaches/uses, etc.)

- Case selection is everything. Identify themoral concepts you wish to cover. Then choose cases that involve these concepts. The debate itself, especially the question and answer session with the judges, can be used to generate a discussion of these concepts.
- The Ethics Bowl is definitely a student-centered activity. It is best for the teacher to assume the role of moderator and intervene only to keep

the discussion focused. If students are properly oriented for the competition, then they assume responsibility themselves for keeping the debate orderly.

- Debriefing is important. Students get plenty of feedback from the competition and need help interpreting it and receiving it constructively. The peer review students also need advice on how to deliver the feedback proactively. We tend to approach the debate from the standpoint of the virtue of reasonableness and provide students with plenty of opportunities to practice this virtue before the competition.

Appendix (Annotated)

Bibliographical Information

- Robert F. Ladenson (2001) "The Educational Significance of the Ethics Bowl". Teaching Ethics 1(1), March 2001: 63-78.
- Jose A Cruz, William J. Frey, and Halley D. Sanchez. (2004) "The Ethics Bowl in Engineering Ethics at the University of Puerto Rico-Mayaguez". Teaching Ethics 4(2), Spring 2004: 15-32.
- Michael Davis (2004) "Five Kinds of Ethics Across the Curriculum". Teaching Ethics 4(2), Spring 2004: 1-14.
- Michael Davis (1998) Thinking Like An Engineer: Studies in the Ethics of a Profession. U.K.: Oxford University Press: 119-156.
- Michael S. Pritchard (1996) Reasonable Children: Moral Education and Moral Learning. Lawrence, KS: University of Kansas Press: 140-163
- James Rest, Darcia Narvaez, Muriel J. Bebeau, and Stephen J. Thoma (1999) Postconventional Moral Thinking: A Neo-Kohlbergian Approach. Mihway, New Jersey: Lawrence Erlbaum Associates, Publishers: 104.
- Mark Johnson (1993) Moral Imagination: Implications of Cognitive Science for Ethics. Chicago, IL: University of Chicago Press: 8-9.
- Jose A Cruz and William J. Frey (2003) "An Effective Strategy for Integrating Ethics Across the Curriculum in Engineering: An ABET 2000 Challenge" Science and Engineering Ethics 9(4): 546-548.

- Chuck Huff and William Frey (2005) Moral pedagogy and Practical Ethics, Science and Engineering Ethics, 11(3): 389-408.

Additional information or annotations for instructors regarding the Student Module Appendix

Editing Module Contents
This module explains the various methods available to edit a module's contents.

The "Edit" tab for the module displays the contents of the "index.cnxml" file. This file contains the text that appears in the module, plus the CNXML instructions to include any image, sound, or other media files that appear in the module. The default content display and edit is with the Edit-In-Place editor. Alternatively, you can display and edit the content with the Full Source editor by clicking `Switch to Editing Full Source`. From this tab you can perform the following actions on the file:

- Import content into the index.cnxml file
- Export the index.cnxml file to an external XML editor
- Edit the index.cnxml file with the Edit-In-Place editor
- Edit the index.cnxml file with the Full Source editor

[missing_resource: /content/m19610/latest/module-edit-in-place.png]

The "Edit" tab displaying contents of the "index.cnxml" file in the Edit-In-Place editor mode.

Import Content into the "index.cnxml" File

To import content into the "index.cnxml" file, use the following steps:

1. Select the format of file you want to import from the drop-down list next to the `Import` button. The valid formats are: Microsoft Word, OpenOffice Writer, XMLSpy/Authentic, and Plain XML.
2. Click `Import`. A screen displays with a text box for the pathname of the file you want to import.
3. Type the pathname of the file in the text box or use `Browse` to display the name of the pathname in the text box.
4. Click `Import`. The content of the file you specified is copied into the "index.cnxml" file in your module.

Note: When you click `Import` in the last step, the contents of the "index.cnxml" file are overwritten by the contents of the imported file.

Export the "index.cnxml" File to an External XML Editor

You can export the "index.cnxml" file so that you can edit it with a software package that is specifically designed to edit XML files. The "index.cnxml" file must be altered to allow the XML editor to read it. And once you have completed your edits, the file must be altered again to remove the changes for editor readability. The Connexions Export and Import functions automatically add and remove the changes for editor readability. Currently, the only XML editor format supported by Connexions is the format used by the Altova **xmlspy**® editor or the Altova **authentic**® editor.

To export the "index.cnxml" file to be edited with an XML editor, use the following steps:

1. Select the format of the file you want to export from the drop-down list next to the `Export` button. The valid formats are: XMLSpy/Authentic, and Plain XML.
2. Click `Export`. A dialog box displays asking if you want to open the file or save it to a disk.
3. Specify that you want to save the file to a disk.
4. Specify the pathname of where you want to save the file.
5. Launch your XML editor and edit the file.

To import the "index.cnxml" file back into Connexions, use the instructions for Import Content from Outside Connexions mentioned earlier in this module.

Edit the index.cnxml file with the Edit-In-Place Editor

The Edit-In-Place editor is the default editor on the "Edit" tab. You can insert new text and media objects into a module or modify the existing

contents of a module with this editor. You do not have to be familiar with the CMXML tags to use Edit-In-Place. It inserts the appropriate CNXML tags for you when you insert a new item in the file. In addition, it has a help feature that describes the CNXML tags for the items you insert or edit with it.

Inserting Content in an Empty Module

To insert text and media objects into a new module that contains no content, use the following steps:

1. Click in the white box that says "Click To Edit Paragraph" in the Edit-In-Place editor.
   [missing_resource: /content/m19610/latest/eip-click-to-insert-text.png]

   The "Click To Edit Paragraph" text box in the Edit-In-Place editor.

   The white box is replaced by a blue editing box.
2. Type the text you want to add in the blue editing box.
   [missing_resource: /content/m19610/latest/eip-edit-empty-paragraph.png]

   The editing box in the Edit-In-Place editor.

3. Click Save to save the text you typed. The blue editing box is replaced by a white box that displays your entry.

Editing Existing Content with Edit-In-Place

To edit the content of a module, use the following steps:

1. Display the module you want to edit with Edit-In-Place.
2. Scroll down to display the white text box that contains the item you want to edit.
3. Click in the text box. The white box is replaced by a blue editing box that displays the text of the element.

[missing_resource: /content/m19610/latest/eip-edit-existing-paragraph.png]

The blue text box in the Edit-In-Place editor.

4. Make the necessary changes to the item.
5. Click `Save` to save the entry or click `Cancel` to clear the entry in the blue editing box.
6. Repeat steps 2 through 5 for any item you want to edit.

Note: In the white text boxes mathematical equations are displayed as they appear in the module. In the blue editing box mathematical equations are displayed in the MathML markup language. Edit equations by making changes to the MathML.

Adding Content Items with Edit-In-Place

You can add new content items to a module with Edit-In-Place. The types of items you can add are: sections, paragraphs, examples, exercises, notes, lists, code blocks, equations, and tables. Edit-In-Place places the opening and closing CNXML tags around the item and it generates a unique item ID for each item you add. To add new items, use the following steps:

1. Display the module you want to edit with Edit-In-Place.
2. Scroll down to display the location in the module in which you want to insert the content item. You can insert the new item above or below and existing item by using the `Insert...` link above or below the existing item

3. Select and click the type of item you want to enter from the drop-down list next to the `Insert...` link. These items are described in the help text, which can be accessed by selecting "Other Elements" from the drop-down list.
4. An empty blue editing box displays for the item you selected. You can display help text for the item you selected by clicking `Help editing xxx` in the upper right corner of the blue box, where xxx is the CNXML tag for the item you selected.
5. Type the text that is appropriate for the type of item in the blue editing box.
6. Click `Save` to save the entry or click `Cancel` to clear the entry in the blue editing box.
7. Repeat steps 2 through 6 for each new content item you want to add to the module.

Edit the "index.cnxml" File with the Full Source Editor

You can edit the content and CNXML tags in the "index.cnxml" file within Connexions with the Full Source editor. This method of editing is recommended for quick changes only. This is a simple text editor and it does not have the advanced functions (for example, spell checking and text searching) that are available with commercially available text editors. To edit the file with the Full Source editor, use the following steps:

1. Click `Switch to Editing Full Source` on the "Edit" tab. The contents of the "index.cnxml" file are displayed in the Full Source editor mode.

[missing_resource: /content/m19610/latest/module-edit-full-source.png]

The "Edit" tab displaying the contents of the "index.cnxml" file in the Full Source editor mode.

2. Scroll down to display the location in the module in which you want to edit.
3. Click in the text window and begin editing.

4. Click Save to save to save your changes.

As you edit the "index.cnxml" file, click Save at any point to save your changes. The page reloads and validates your CNXML markups every time you save your changes. If there are no CNXML tag errors, your markup is valid and "Saved" appears in an information box at the top of the tab. If there are errors in your CNXML markup, they are listed above the file content.

Note: Saving and publishing are two different operations. When you save your editing changes in the Full Source editor, your changes are saved in your work area; they are not entered into the Connexions repository. Your module must be published to be entered into the repository and be accessible by visitors to Connexions. For information about publishing, see the topic Publishing the Module later in this module.

Validating Your Work in the Full Source Editor

Any "index.cnxml" file that you publish in the Connexions repository must be a valid CNXML document. This means that the file must be well formed and contain no CNXML errors. If it does contain errors, a description of each error appears when you click Save. Your changes are not saved until the errors are corrected.

[missing_resource: /content/m19610/latest/module-edit-full-source-errors.png]

Example of validation errors in the Full Source editor

A description of the errors appears in the information box that appears above the content.

Note: You cannot publish the module in Connexions until all the CNXML errors are corrected and the "index.cnxml" file is validated.

Instructor Module English--Three Frameworks for Ethical Decision-Making and Good Computing Reports

This instructor module provides pedagogical commentaries on teaching the student module, Three Frameworks for Ethical Decision-Making and Good Computing Reports (m13757). It is bilingual with Spanish commentary provided by Ramon Ramos-Chevres and rough English translations provided by William Frey. It and the corresponding student module have been developed as a part of an NSF-funded project, "Collaborative Development of Ethics Across the Curriculum Resources and Sharing of Best Practices," NSF SES 0551779.

Instructor Module Template

Student Module Title: Three Frameworks for Ethical Decision Making and Good Computing Reports m13757

Introduction/ Summary

I. Summarize the student module. Include the content objectives and skill objectives of the student as well as the exercises it features.

Este módulo le provee al estudiante herramientas que va a utilizar en el proceso de tomar decisiones basadas en el concepto de la ética. Una vez que los estudiantes tuvieron la oportunidad de adquirir el módulo, de estudiarlo y de recibir explicación del profesor acerca de los objetivos, metas e instrucciones, estarán preparados para comenzar un proceso de análisis junto con el profesor. Los estudiantes analizarán distintos casos por medio de esta herramienta cumpliendo con el objetivo principal que es integrar la ética en su proceso de tomar decisiones. Se deberá aplicar en el análisis, los tres marcos de referencia: división del proceso de tomar decisiones, proceso de probar las soluciones y la prueba de viabilidad.

This module provides the student with tools that can be used to make ethical decisions. Students read the module and then attend a formal presentation given by the instructor that outlines the four stages of decision-making. This helps prepare them for case-based, decision-making exercises carried out with feedback provided by the instructor. When they use the three frameworks discussed in the module to analyze case studies, they

succeed in integrating ethics into the decision-making process. Of the three frameworks discussed in the module, one condenses decision making into four stages (problem specification, solution generation, solution testing, and solution implementation), another offers three tests to validate the ethics of solution alternatives (reversibility, harm/benefits, and publicity), and a third covers solution implementation by examining obstacles that arise during this phase (such as resource, interest, and technical constraints).

II. Preparing the environment/atmosphere. In this section, discuss what can be done to create an environment or atmosphere conducive to bringing about the learning objectives of the student module. This could include background readings, advance home work, pre-module writing, activities to get students accustomed to discussing, and so forth.

Con el objetivo de crear un ambiente de discusión y análisis de este modulo, se recomiendan estrategias para lograr el interés del estudiante. Como introducción al modulo, se podría asignar una tarea donde el estudiante deba analizar un caso y tomar una decisión utilizando como herramientas "sus experiencias individuales". De manera que, sea el propio estudiante que utilice juicios valorativos y se acerque al uso del concepto de la ética y toma de decisiones. Una vez captada esa atención del estudiante, se le provee la información necesaria para el análisis basado en el modulo: contenido del modulo, ejemplos y discusiones de casos, recursos externos como libros, periódicos, revistas y material que cumpla con los objetivos del modulo. Se espera que el estudiante asuma una posición de discutir y debatir ideas en un ambiente amigable y de confianza. Se espera que esto facilite el aprendizaje, por lo que el profesor/ a debe:

Several strategies can be used to create an environment conducive to discussion and analysis. To introduce the module, the teachers can assign students a case and have them informally analyze it using their experience as a frame of reference. Getting students used to making value judgments helps them to practice the different skills involved in ethical thinking and decision-making. Such pre-module homework awakens the students' interest in ethics and can form a basis upon which the teacher can build by providing more information, examples of ethical problems, case

discussions, external resources,(books, newspapers, magazines, movies, novels) and other resources that promote ethical decision-making. Moreover, this module gives students practice adopting and defending a position in a civil and confident manner.

- **Dar la oportunidad a que el estudiante hable libremente haciendo preguntas y conversando con los estudiantes de una forma organizada.**
- **Escuchar con atención a cada integrante del grupo y dar su interpretación.**

- **Estructurar un dialogo con el resto de la clase.**
- **Permitir interpretaciones de otros estudiantes.**
- **Identificar aquellas respuestas que cumplen con los objetivos del modulo y publicarla ante todos.**
- **Estar disponible y preparado/a para aclarar dudas.**

Suggestions to Facilitate Learning

- Give students the opportunity to speak freely, ask questions, and discuss issues with classmates in an orderly manner.
- Listen to other speakers attentively and actively interpret what they hear.
- Hold a structured dialogue with the rest of the class.
- Permit interpolations and comments from other students.
- Identify and underline student responses that advance module objectives by instantiating ethical principles and concepts.
- Be open to answering questions and responding to doubts.

III. Learning Objectives: Choose one or both of the following set of moral learning objectives. (The UPRM objectives map almost completely on the Hastings Center objectives.)
UPRM Ethical Objectives

- **Ethical Awareness**: Students are able to recognize and characterize the ethical issues that arise in ordinary situations. In pre test format, student recognizes one ethical issue embedded in a realistic scenario.

- **Ethical Integration**: Students use ethical considerations as specifications in "designing" a solution to a realistic ethical problem. They design solutions to ethical problems that properly and adequately respond to ethical considerations or ethics tests.
- **Preventive Ethics**: Students uncover a latent or potential ethical problem and design effective counter measures. They use sociothecnical analysis or other "tool" to find value conflicts that arise with the implementation of a new product or service and design countermeasures to defuse the conflict.
- **Value Integration**: Students find opportunities in realistic situations for realizing moral value through the exercise of their technical and occupational skills. In consultation with a local community, they identify a need and develop means to respond to this need; usually done in service learning context.
- **Ethical Evaluation**: Students use ethical considerations (approaches, tests) to evaluate and rank alternative solutions to a realistic decision – eliciting situation. In a Gray Matters activity, students correctly rank the alternatives in response to an ethics dilemma or situation.

Hasting Center Skills

- **Moral Imagination**: Module allows students to practice and develop their moral imaginations. Students successfully employ reversibility test to view action alternative from the standpoint of one of the action's targets.
- **Moral Sensitivity**:Module raises students' sensitivity to the moral issues that arise in everyday situations. Students successfully identify moral issues embedded in scenario in pre-test or Gray Matters format.
- **Moral Analysis**: Students practice and develop ability to analyze moral concepts and principles. They successfully employ moral approaches and principles in debate of moral issues. (Ethics Bowl or chair debate)
- **Moral Responsibility**: Module elicits from students a sense of moral responsibility. They employ concepts in moral responsibility to assign praise or blame in a moral case or scenario.
- **Tolerating ambiguity and disagreement**: Module helps students tolerate moral ambiguity and disagreement while striving for moral

clarity and agreement. Students debate a difficult and controversial moral case with clarity and civility.

IV. Relationship of student module to ethics requirements of targeted accreditation effort AACSB

- **Ethical Leadership**: Helping students to "see the criticality of ethical leadership to effective and successful management."
- **Ethical Decision-Making**: Learning experiences should expose students to cases and types of ethical issues that they are likely to face in the business world—both to enhance their abilities to recognize ethical issues and to increase their ethical sensitivity and awareness."
- **Responsibility of Business in Society**: Students should "understand the symbiotic relationship between business and society, especially in terms of the moral dimensions of the power placed in the hands of owners and managers."

Corporate Governance

- Role and responsibilities of board of directors and audit committee.
- Internal controls and role responsibilities of management.
- Monitoring activities such as internal auditing.
- Element of an effective code of conduct.
- U.S. Federal Sentencing Guidance and Sarbanes-Oxly.
- Components of an effective corporate compliance program
- Role responsibilities of public accountants, counsel, and regulatory bodies (EETF 14)

Source: Ethics Education in Business Schools: Report of the Ethics Education Task Force to AACSB Internal's Board of Directors Relation To Acreditation Effort: ABET

- **Criterion 3c**: Ethics (and others components) in design.
- **Criterion 3d**: Multidisciplinary Team Skills.
- **Criterion 3f**: Professional Ethical Responsibility.
- **Criterion 3h**: Understanding global and social impacts of engineering.

V. Pedagogical Strategies (This list is, by no means, exhaustive. Space provided for strategies not mentioned previously)

- **Lecture**: Providing summaries and explanation of materials in form of formal or informal class presentation by teacher.
- **Reading / Pre-module Assignments**: Students are assigned readings and other activities to help them prepare for module.
- **Writing (Formal and Informal / Individual and Group)** : Students prepare individually or in groups formal or informal written responses to module content.
- **Discussion (Class and Group / Formal and Informal)**: Teachers pose questions to student for discussion. Students give formal or informal presentations.
- **Individual or group activities**: Students are divided in groups to carry out different exercises. Students carry out exercises individually.
- **Debate**: A form of discussion where students take and defend a position.
- **Framework-Driven**: Module allows for practicing frameworks (heuristics like decision-making procedures) or tests (ethical approaches, ethics tests)
- **Identifying and reinforcing "good move" moments**: Teacher underlines / emphasizes student comments that advance the module's learning objectives.

VI. Assurance of Learning ("Tools" and Descriptions)

- **Muddiest Point**: Students are asked to discuss or write on the parts of the module they found most difficult.
- **Two Minute Paper (Informal class or out of class "writes")**: Students write a short essay responding to and reflecting on the module and learning experience.
- **Discussion Points**: A formal or informal discussion is held with the class (or groups) reflecting on the module learning experience.
- **Observer/Assessor Observations**: An outside observer attends class while the module is presented and provides feedback.
- **Feedback (Evaluation) Forms**: Forms designed to elicit feedback on the module. (For example: forms that have students rate and rank

different modules.)

- **Connexions® EAC Toolkit Instructor Module**: Assessment experiences can be integrated to develop a Toolkit Instructor Module.
- **Formal Evaluation Forms (Student and Peer)**: Forms used by department or university to gather formal evaluation data.

TI DSP/BIOS QUE Module
This module describes the basics of the TI DSP/BIOS v5.x Queue (QUE)
Module.

Introduction

This module describes the basics of the Queue (QUE) module in DSP/BIOS
v5.x. It also shows how to use the MEM module to allocate memory for
queue elements.

Reading

- SPRU423 TMS320 DSP/BIOS Users Guide: Read the section title
  Queues
- SPRU403 TMS320C6000 DSP/BIOS 5.x Application Programming
  Interface (API) Reference Guide: Reference the sections on QUE and
  MEM

QUE Module

The QUE module provides a way to manage linked lists of data objects. A
linked list is a group of objects where an object in the list has information
about the previous and next elements in the list as demonstrated in Figure 1.
A linked list can be used to implement buffers that operate in a FIFO, LIFO,
LILO and FILO manner, to name a few. Many times it is used in a FIFO
manner.
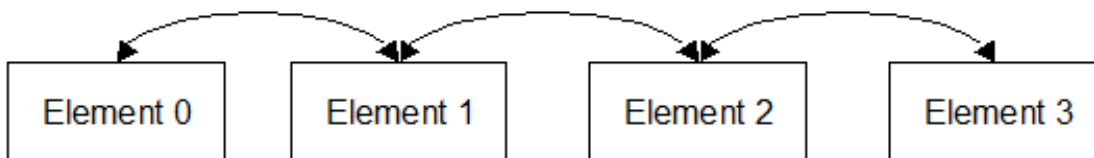
Diagram of a linked list

The basic element in a queue structure is a `QUE_Elem` which is defined by

```
typedef struct QUE_Elem {
    struct QUE_Elem *next;
    struct QUE_Elem *prev;
    } QUE_Elem;
```

This definition is in the header file `que.h` which must be included in the source file.

A message or object put on a queue is a structure that has as its first element a `QUE_Elem`. The message is defined by the user's code. An example would be:

```
typedef struct MsgObj {
    QUE_Elem elem; /* first field for QUE */
    Int val; /* message value */
    } MsgObj, *Msg;
```

To put a message at the end of a queue use the function `QUE_put` and to remove a message from the beginning of the queue use `QUE_get`. These two functions allow the queue to be used as a First In First Out (FIFO) buffer.

Here is another example of a message structure. Notice that you can put anything else in the struture for your message as long as the QUE_Elem element is the first in the structure.

```
typedef struct MsgObj {
    QUE_Elem elem; /* first field for QUE */
    float array[100]; /* message array */
    } MsgObj, *Msg;
```

When creating messages it may be necessary to get a segment of memeory to store the message. The following code can be used to allocate memory for a message and put it on a queue.

```
Msg msg; // pointer to a MsgObj, this doesn't
allocate memory for a message

    msg = MEM_alloc(0, sizeof(MsgObj), 0); //
Allocate memory for the object
    if (msg == MEM_ILLEGAL) {
    SYS_abort("Memory allocation failed!\n");
    }
    /* fill the message with data here */
    /* This puts the address of the message in the
queue */
    QUE_put(&queue, msg);
```

In order to use the MEM module the header file `mem.h` must be included in the source file. The `MEM_alloc` gets a segment of memory from the memory manager and returns a pointer to the segment. The address gets put into the `msg` pointer so that when we access the object it points to segment of memeory.

`QUE_put` puts the address of the message object on the queue. This function only copies the message pointer in order to minimize the amount of processing needed to put the message on the queue.

Before trying to get a message from a queue, the code should check to see if there is a message available on the queue. The following code makes this check and then gets the message from the queue.

```
if (QUE_empty(&queue)) { // Check to see if the
queue is empty
    LOG_printf(&trace,"queue error\n"); // Print
something if it is empty
    // If the queue is empty you probably will not
want to proceed
```

```
    }
    msg = QUE_get(&queue); // If there is a
message, dequeue it
    // use the message here
    // After using the message, free the memory
    // This tells MEM_free the location of the msg
and size so it
    // can remove it from memory
    MEM_free(0, msg, sizeof(MsgObj)); // Free up
the memory
```

Functions QUE_put and QUE_get are atomic in that they add and remove elements from the queue with interrupts turned off. Therefore there should not be a problem of more than one task trying to access the queue at the same time. The function QUE_get is also non-blocking so the tasks should determine if there are any elements on the queue before calling QUE_get.

A semaphore can be used to count the number of elements on a queue and be used to block a task that needs access to a queue. Figure 2 shows how one task TSK0 will write to a queue, QUE0, and the other task TSK1 will read from the queue. Also, the semaphore QUE0_SEM is used to keep track of how many elements are on the queue. After TSK0 puts a message on the queue it will call SEM_post and increment the semaphore. Before reading from the queue, TSK1 will call SEM_pend on the semaphore and if the task does not block, there is an element on the queue. If there are no elements on the queue, the task will block on the semaphore.
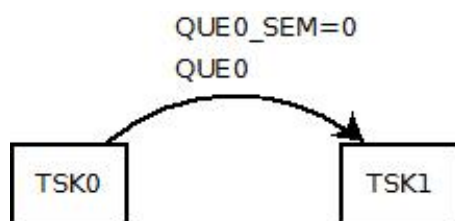
Queue and semaphore
setup

For the setup with one queue and one semaphore notice that if the program ran for a long time the code would have to continually allocate memory for a message and then de-allocate it when it was done using the message. This could take up a substantial amount of time and could cause fragmentation of the memory space. A better method is to have two queues where one queue holds messages that are free and one holds messages that contain data being transmitted from one task to another. Figure 3 shows the same setup in Figure 2 except there is now a queue that contains free messages or empty messages.

QUE0_SEM=0
QUE0

TSK0   TSK1

QUE_Free_SEM=10
QUE_Free
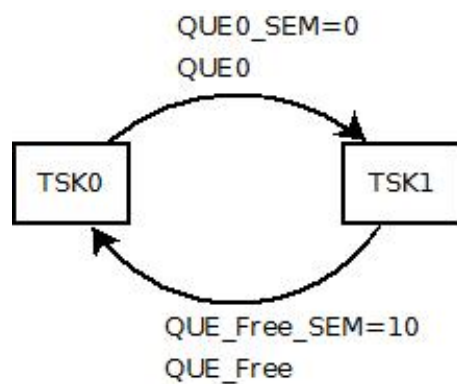
Free message queue
setup

During the initialization phase of the program, memory for empty messages is allocated and the messages are put in the `QUE_Free` queue and the semaphore `QUE_Free_SEM` is incremented for each message put on the queue. When `TSK0` needs to send a message to `TSK1` it will check `QUE_Free_SEM` to see if there are any free messages. If not, it will block. If there are free messages it will take one off of `QUE_Free` after the `QUE_Free_SEM` is decremented and then fill the message with data and

put it on the queue QUE0. The semaphore QUE0_SEM is incremented after the message is put on QUE0.

The task TSK1 will block on QUE0_SEM until task TSK1 puts a message on the queue. Then it will decrement the semaphore and use the message. When it is done it will put the message on QUE_Free and increment its semaphore.

Example

This is a very simple example to demonstrate the structure of a program that uses queues. In the example we will assume that DSP/BIOS has been set up with two tasks, TSK0 and TSK1, and one queue, QUE0. The two tasks have the same priority and TSK0 is set to execute first. The code for the example follows.

```
#include <std.h> // Target definition header
    #include <sys.h> // DSP/BIOS config/error
header
    #include <log.h> // LOG module header
    #include <mem.h> // MEM module header
    #include <que.h> // QUE module header
    #include <tsk.h> // TSK module header

    #include "QUE_Examplecfg.h" // header
generated by QUE_Example.tcf config file

    typedef struct MsgObj {
    QUE_Elem elem; /* first field for QUE */
    Int val; /* message value */
    } MsgObj, *Msg;
    Void main()
    {
    }
    // TSK0 will generate two messages and put
them on the queue
    Void funTSK0()
```

```c
{
Msg msg; // Pointer to the message object
// allocate memory for first message
msg = MEM_alloc(0, sizeof(MsgObj), 0);
if (msg == MEM_ILLEGAL) {
// If the memory allocation fails, abort
SYS_abort("Memory allocation failed!\n");
}
msg->val = 1; // put the message number in the
message
// print the message number
LOG_printf(&trace, "Writing message %d", msg->val);
// Put the message on the queue
QUE_put(&QUE0, msg);


// repeat for the second message
msg = MEM_alloc(0, sizeof(MsgObj), 0);
if (msg == MEM_ILLEGAL) {
// If the memory allocation fails, abort
SYS_abort("Memory allocation failed!\n");
}
msg->val = 2; // put the message number in the
message
// print the message number
LOG_printf(&trace, "Writing message %d", msg->val);
// Put the message on the queue
QUE_put(&QUE0, msg);
}
// TSK1 will get two messages from the queue
Void funTSK1()
{
Msg msg; // Pointer to the message object

// If the queue is empty, we should not
```

```
proceed
     if (QUE_empty(&QUE0)) {
     LOG_printf(&trace,"TSK1 queue error");
     return; // This will make the task terminate
     }
     // Get the message off the queue
     msg = QUE_get(&QUE0);
     // print value in the message
     LOG_printf(&trace, "Reading message %d", msg->val);
     // Since we are done with the message, free
the memory
     MEM_free(0, msg, sizeof(MsgObj));


// Repeat for the second message
     // If the queue is empty, we should not
proceed
     if (QUE_empty(&QUE0)) {
     LOG_printf(&trace,"TSK1 queue error");
     return; // This will make the task terminate
     }
     // Get the message off the queue
     msg = QUE_get(&QUE0);
     // print value in the message
     LOG_printf(&trace, "Reading message %d", msg->val);
     // Since we are done with the message, free
the memory
     MEM_free(0, msg, sizeof(MsgObj));
     }
```

It is important to free the memory of each message after it is used so that
the memory does not get used up. The result of the run follows.

```
Writing message 1
    Writing message 2
    Reading message 1
    Reading message 2
```

Building an Ethics Module for Business, Science, and Engineering Students

Pre-Test Format in Word
https://cnx.org/content/m15454/

Clicking on this figure
opens an exercise in the
Pre-Test format in Word.

Gray Matters Format
https://cnx.org/content/m15454/

This figure provides in
Word a Gray Matters
format used to present key
decision points in the
Hughes Aircraft case. (See
Computing Cases for more
on this case.)

Introduction

Sample Introduction
This module is designed to help you understand how ethical issues arise daily in your field. You will examine everyday scenarios or decision points and respond in terms of the ethical issues that arise. Below are frameworks that describe how to make ethical decisions, how to solve problems with

ethical implications, and how to test your decisions and solutions in terms of their ethics.

What you need to know …

Sample description of module content
This section provides general background information. It includes information on how to (1) define problems, (2) design and evaluate ethical solutions, and (3)resolve disagreements. These frameworks can be used with the Pre-Test and Gray Matters activities.

Problem Solving Stages (Based on analogy between the problem solving and design processes

1. **Problem Specification or Definition:** This stage consists of defining the problem you face from different standpoints or frames. Carefully defining your problem is an essential step to designing effective and ethical solutions. Defining your problem from multiple frames or vantage points, also helps you to create imaginative and ethical solutions to problems that appear unsolvable under commonplace framings.
2. **Solution Generation:** In this stage, you will try to resolve the problem you defined in the previous stage. In a section below, you will find a list of generic solutions to disagreements between stakeholders.
3. **Solution Testing:** The solutions developed in the second stage must be tested in different ways. The **reversibility test** encapsulates the ethical theory of deontology; exploring the issue from the standpoint of those on the receiving end of your action outlines the idea of reciprocity which is fundamental to deontology. The **harm/benefits** test has you weigh benefits against harms and steers you toward that solution that produces the most benefits and the least harms. This provides a reasonable approximation to the theory of Utilitarianism which enjoins us to produce the greatest good for the greatest number. Finally, the **publicity test** has you attribute the values embedded in the act to the character of the agent. in this way, the publicity test encapsulates virtue ethics.
4. **Solution Implementation:** The chosen solution must be examined in terms of how well it responds to various situational constraints that

could impede its implementation. To carry out this stage, imagine a check list of resource, interest, and technical constraints that could give rise to obstacles. Go through the list to see if any of these constraints applies to your solution.

Problems can be defined in different ways. By looking at a problem through different definitional frames, we are able to uncover non-obvious, creative solutions.

1. **Technical Puzzle:** If the problem is framed as a technical puzzle, then solutions would revolve around developing designs that optimize both ethical and technical specifications, that is, resolve the technical issues and realize ethical value. In this instance, the problem-solver must concentrate on the hardware and software components of the surrounding socio-technical system (STS).
2. **Social Problem:** If the problem is framed as a social problem, then solutions would revolve around changing laws or bringing about systemic reform through political action. This would lead one to focus on the people/groups/roles component (working to social practices) or the legal component of a socio-technical system.
3. **Stakeholder Conflict:** If the problem is framed as a conflict between different stakeholder interests, then the solution would concentrate on getting stakeholders (both individuals and groups) to agree on integrative or compromise-building solutions. This requires concentrating on the people/group/role component of the STS. (Note: A stakeholder is any group or individual with a vital interest at play in the situation.)
4. **Management Problem:** Finally, if the problem is framed as a management problem, then the solution would revolve around changing an organization's procedures. Along these lines, it would address the organization's (1) fundamental goals, (2) decision recognition procedures, (3) organizational roles, and/or (4) decision-making hierarchy. These are the four components comprise the organization's CID (corporate internal decision) structure.

Ethics Tests

1. **Reversibility:** Would this solution alternative be acceptable to those who stand to be most affected by it? To answer this question, change places with those who are targeted by the action and ask if from this new perspective whether the action is still acceptable?
2. **Harm / Benefits:** What are the harms your solution is likely to produce? What are its benefits? Does this solution produce the least harms and the most benefits when compared to the available alternatives?
3. **Publicity:** Would you want to be publicly associated or identified with this action? In other words, assume that you will be judged as a person by others in terms of the moral values expressed in the action under consideration. Does this accord with how you would aspire to be judged?

One of the most difficult stages in problem solving is to jump start the process of brainstorming solutions. If you are stuck then here are some generic options guaranteed to get you "unstuck."

1. **Gather Information:** Many disagreements can be resolved by gathering more information. Because this is the easiest and least painful way of reaching consensus, it is almost always best to start here. Gathering information may not be possible because of different constraints: there may not be enough time, the facts may be too expensive to gather, or the information required goes beyond scientific or technical knowledge. Sometimes gathering more information does not solve the problem but allows for a new, more fruitful formulation of the problem. Harris, Pritchard, and Rabins in Engineering Ethics: Concepts and Cases show how solving a factual disagreement allows a more profound conceptual disagreement to emerge.
2. **Nolo Contendere.** Nolo Contendere is latin for not opposing or contending. Your interests may conflict with your supervisor but he or she may be too powerful to reason with or oppose. So your only choice here is to give in to his or her interests. The problem with nolo contendere is that non-opposition is often taken as agreement. You may need to document (e.g., through memos) that you disagree with a course of action and that your choosing not to oppose does not indicate agreement.

3. **Negotiate.** Good communication and diplomatic skills may make it possible to negotiate a solution that respects the different interests. Value integrative solutions are designed to integrate conflicting values. Compromises allow for partial realization of the conflicting interests. (See the module, The Ethics of Team Work, for compromise strategies such as logrolling or bridging.) Sometimes it may be necessary to set aside one's interests for the present with the understanding that these will be taken care of at a later time. This requires trust.
4. **Oppose.** If nolo contendere and negotiation are not possible, then opposition may be necessary. Opposition requires marshalling evidence to document one's position persuasively and impartially. It makes use of strategies such as leading an "organizational charge" or "blowing the whistle." For more on whistle-blowing consult the discussion of whistle blowing in the Hughes case that can be found at computing cases.
5. **Exit.** Opposition may not be possible if one lacks organizational power or documented evidence. Nolo contendere will not suffice if non-opposition implicates one in wrongdoing. Negotiation will not succeed without a necessary basis of trust or a serious value integrative solution. As a last resort, one may have to exit from the situation by asking for reassignment or resigning.

What you will do ...

Sample description of module activities
In this section you will explore different activities designed to give you practice in identifying ethical issues in real world situations, carrying out ethical analysis, designing solutions to ethics problems, and implementing ethical solutions over situational constraints.

Sample instructions to students - Pre-Test for Introduction to Computers
Read the following scenario (Step 1). Then individually or in groups carry out steps two through three.
Step 1: Individually evaluate the scenarios below using the following three questions:

- Do you think this situation is common/realistic? Yes or No
- Do you think this situation or activity is Ethical or Unethical?
- In general would others agree with your answer to Q #2? Yes or No

Betting Pool: While reviewing e-mail messages a manager discoves someone using the company's e-mail system to operate a weekly betting pool.

- Do you think this situation is common/realistic? Yes or No
- Do you think this situation or activity is Ethical or Unethical?
- In general would others agree with your answer to Q #2? Yes or No

Step Two: Informally share or discuss your answers with the class. Use the space below to make notes.

_____
__
_____
__
_____
__

Step Three: Use these tests to structure your discussion of another scenario.

- **REVERSIBILITY**: Would I think this a good choice if I were among those affected by it?
- **PUBLICITY**: Would I want this action published in the newspaper?
- **HARM**: Does this action do less harm than any available alternative?
- **FEASIBILITY**: Can this solution be implemented given resource, interest, and technical constraints?

Sample instructions to students - Gray Matters for Business Ethics
Gray Matters is based on an exercise developed by Lockheed Marietta. Used by companies like Boeing, it consists of evaluating and ranking solutions to the ethical problems raised in short, decision points. Your job is to read the scenario below, review the solution alternatives, evaluate them in terms of ethics tests, and choose the best.

Gray Matters

Pacemaker Case:
A pacemaker manufacturing company (PACE Inc.) located in a small town in Puerto Rico provides jobs to about 80% of the town's workforce. Profit margins are thin in this competitive field which includes larger U.S. companies. You are on an R and D team for PACE that has studied two options for the circuitry: BULK CMOS and SOI. The team favors BULK CMOS because the manufacturing process is simpler and cheaper. But the chips will be larger and consume more energy; this means more surgery for the patients to replace the batteries. Overall, the use of BULK CMOS would reduce patient life expectancy by 15%. Given this knowledge, what should you do?

Solution Alternatives

- Go along with the team and advocate the simpler and cheaper process.
- Oppose the team and advocate the more complex, more expensive, but safer process. Try to persuade the team members to opt for safety.
- Oppose the team. Force agreement by threatening to blow the whistle.
- Resign from PACE, Inc.
- Design your own solution.

What did you learn?

This section provides closure to the module for students. It may consist of a formal conclusion that summarizes the module and outlines its learning objectives. It could provide questions to help students debrief and reflect on what they have learned. Assessment forms (e.g., the "Muddiest Point" Form) could be used to evaluate the quality of the learning experience. In short, this section specifies the strategy for bringing the module to a close.

Sample Module Close Out
Reflection helps us successfully to close the act of learning. Module activities are designed to give us feedback on our decisions and problem solving. How did your group, your class, and your teacher react to your conclusions and arguments? What can you learn from these reactions?

Sample Closing Exercise

- Reasonableness consists of seeking value integrative solutions to ethical problems, drawing compromises without sacrificing integrity, being open to the ideas of others, and providing careful explanations and justifications of your own ideas. Evaluate yourself and your classmates on how well you realized this virtue during this exercise
- Polarization of positions, interests, and people is one of the biggest blocks to problem solving in ethics. It consists of dividing solutions, interests, positions, and people into two camps, one (those who agree with you) being absolutely good, the other (those who disagree with you) being absolutely bad. Did you see evidence of polarization during this activity?
- Give your teacher some feedback here. What was the strongest point of this exercise? The weakest point?
- Finally, list five things that you learned as a result of this exercise.

Appendix

This optional section contains additional or supplementary information related to this module. It could include: assessment, background such as supporting ethical theories and frameworks, technical information, discipline specific information, and references or links.

Useful EAC Links

Included with this module are a series of links designed to connect students with materials available online that provide background information in ethics. Review them, keep the links useful for your module, and delete those that go beyond your module's scope.

- The glossary available through Online Ethics provides concise and relatively noncontroversial definitions of terms important to ethics
- The link to Computing Cases provides a general orientation to the ethics tests used in the pre-test and gray matters exercises. Computing Cases also contains materials from six extensive case study analyses that have been prepared with funds from the NSF.

- Gary Comstock's NSF-funded project, Langure, provides browsers with several modules on research ethics that can be combined into different courses. Comstock has prepared an excellent introduction to the ethical theories of utilitarianism, deontology, and virtue ethics. This is useful for teachers who want to take a more theoretical approach to teaching their ethics module.
- Ethics Updates, a website developed and maintained by Larry Hinman from the University of California, San Diego, provides excellent materials and links in areas such as applied ethics, theoretical ethics, and ethics across the curriculum. The theoretical ethics section has links to several classical philosophical texts in ethics that are publicly available online.
- The EAC Module Collection, the last link, draws together resources available on Connexions for developing, assessing, and disseminating EAC modules.

EAC ToolKit Project

This module is a WORK-IN-PROGRESS; the author(s) may update the content as needed. Others are welcome to use this module or create a new derived module. You can COLLABORATE to improve this module by providing suggestions and/or feedback on your experiences with this module.

Please see the [Creative Commons License](#) regarding permission to reuse this material.

EAC Toolkit - Instructor Module Template

Word Version of this Template
https://cnx.org/content/m14290/

This is an example of an embedded link. (Go to "Files" tab to delete this file and replace it with your own files.)

Instructor Information (Sharing Best Practices in EAC)

This section contains information related to the corresponding Student Module. The intent and expectation is that this information will evolve based on the experiences and collaborations of authors and users. The following categories will sort out this information as authors, collaborators and users experiment with the modules.

Note: Provide a link to the corresponding student module in Connexions (cnx.org) and/or references to relevant …

```
- modules or resources available online
- textbook cases or exercises
- magazine or journal articles
- news stories
- movies, programs or plays
- etc.
```

Module-Background Information

Source, condition and use-history of the module

Learning Objectives

Objectives and outcomes for the module including references to accreditation criteria or standards

Instructional / Pedagogical Strategies

Pedagogical or instructional strategies used or suggested for this module. Examples: lecture, discussion, debate, decision-making, presentation,

dramatization, role playing, cooperative learning, formal or informal writing, etc.

Assessment / Assurance of Learning

Assessment or assurance of learning methodology. Examples: 1-minute paper, muddiest point, quiz/test items, oral presentation, student feed-back, etc.

Pedagogical Commentary

Suggestions for improvements to authors or instructors, questions or comments directed to the EAC community, pitfalls or frustrations, novel approaches including new uses

Other

Anything else?

EAC ToolKit Project

This module is a WORK-IN-PROGRESS; the author(s) may update the content as needed. Others are welcome to use this module or create a new derived module. You can COLLABORATE to improve this module by providing suggestions and/or feedback on your experiences with this module.

Please see the [Creative Commons License](#) regarding permission to reuse this material.

EAC Toolkit - Student Module Template

Word Version of this Template
https://cnx.org/content/m14291/

This is an example of an
embedded link. (Go to
"Files" tab to delete this
file and replace it with
your own files.)

Introduction

This will get your students started; include a brief concise overall view of the module and its objectives broadly stated.

What you need to know …

Include information that you expect your students to study and learn in this module as well as information that will help them carry out the module activities.

What you will do ...

In this section, you will describe the module's activities and/or exercises. You may also provide students with step by step instructions on how to carry them out.

What did you learn?

This section provides closure to the module for students. It may consist of a formal conclusion that summarizes the module and outlines its learning objectives. It could provide questions to help students debrief and reflect on what they have learned. Assessment forms (e.g., the "Muddiest Point" Form) could be used to evaluate the quality of the learning experience. In short, this section specifies the strategy for bringing the module to a close.

Appendix

This optional section contains additional or supplementary information related to this module. It could include: assessment, background such as supporting ethical theories and frameworks, technical information, discipline specific information, and references or links.

EAC ToolKit Project

This module is a WORK-IN-PROGRESS; the author(s) may update the content as needed. Others are welcome to use this module or create a new derived module. You can COLLABORATE to improve this module by providing suggestions and/or feedback on your experiences with this module.

Please see the Creative Commons License regarding permission to reuse this material.